Development Economics of China
2025, VOL. 9, NO. 6, 25-28
DOI: 10.47297/wspdecWSP2515-797305.20250906

A Study on Quantitative Trading Strategies Based on Multimodal Data Fusion and Meta-Reinforcement Learning

Bolai Chong*

Department of Tourism and Hotel Management, Xiamen University, Xiamen, Fujian, 361005, China

ABSTRACT
Quantitative trading often faces challenges such as strategy failure, insufficient data penetration, and limited path adaptability under extreme market conditions. This paper proposes a trading framework that integrates multimodal data—including news sentiment and stock price sequences—with meta-reinforcement learning to systematically enhance the adaptability and robustness of trading strategies. By constructing a hierarchical reinforcement learning system and a multimodal feature fusion network, the framework enables efficient perception and responsive scheduling under complex market dynamics. Experimental results demonstrate that the proposed method achieves superior risk-return performance in both U. S. and Chinese stock markets, with an average return improvement of over 15%. Furthermore, it maintains low volatility and maximum drawdown even under extreme policy shock scenarios, validating its robustness across markets and modalities.

KEYWORDS
Quantitative trading; Reinforcement learning; Multimodal data

1 Introduction

Quantitative trading (FT) is widely used in automated trading systems, aiming for arbitrage and liquidity management with millisecond or even microsecond latency. However, in real-world scenarios, such systems often fail due to extreme market events, noise interference, and the rigidity of strategy parameters. Traditional quantitative strategies primarily rely on structured data (such as OHLC data and technical indicators) and exhibit significant limitations when dealing with unstructured sentiment data. They struggle to establish causal inference pathways and lack the agility to respond to sudden events.

Multimodal data fusion enhances a trading system ' s market perception by incorporating unstructured information such as news sentiment, sentiment scores, and social media comments. In situations like unexpected policy changes or market panic, sentiment often serves as a leading indicator of price movements. Meanwhile, meta-reinforcement learning (Meta-RL), an extension of reinforcement learning, allows for rapid adaptation to new tasks in unfamiliar environments, making it particularly suitable for handling cross-market transfers and multi-task switching in trading scenarios.

This paper aims to construct a meta-reinforcement learning trading framework based on multimodal inputs. The main contributions include: the design of a FinBERT-based sentiment analysis module; the construction of a fusion network to integrate multimodal weights; the implementation of hierarchical control and multi-strategy portfolio optimization; and the introduction of adversarial perturbation testing to enhance robustness.。

2 Literature review

In quantitative trading, single-source data often fails to comprehensively capture market dynamics. In recent years, researchers have begun integrating structured market data with unstructured sentiment/emotion data to improve predictive performance. Common approaches include:

(1) Using Natural Language Processing (NLP) to extract emotional and event information from news articles or social media, then concatenating them with historical price and volume data;

(2) Employing attention mechanisms to allocate weights across multimodal features—for example, aligning news content and price sequences via cross-attention to emphasize key information;

(3) Applying Graph Neural Networks (GNNs) to model stocks and their relationships as graph structures, embedding sentiment and event attributes into nodes or edges, and using graph attention to capture asset-to-asset transmission effects;

(4) Encoding financial events as vectors via event embedding techniques and inputting them into deep networks alongside market sequences to form event-driven prediction models.

Meta-Reinforcement Learning (Meta-RL) aims to enable trading agents to quickly adapt to new market environments through cross-task learning. Compared to training from scratch, Meta-RL leverages prior experience across diverse market

*** Corresponding Author:** Bolai Chong, 3128301552@qq.com

tasks to parameterize a generalized strategy as initialization, which can then be fine-tuned with limited new data. A typical method, MAML, has been applied in index futures trading: researchers categorize different market states into separate tasks, train a transferable initial strategy using meta-learning, and then adapt it to new assets using only a few trading samples. Empirical results show that this approach outperforms conventional deep RL models. This framework effectively addresses the issues of low sample efficiency and poor generalization in deep RL under non-stationary financial environments.

Hierarchical Reinforcement Learning (HRL) decomposes trading decisions into high-level planning and low-level execution, improving manageability and learning efficiency for complex tasks. The high-level agent performs macro-level scheduling among multiple sub-strategies—for example, selecting strategies with different risk preferences or frequencies based on trend signals. The low-level agents focus on concrete actions such as timing and sizing of orders. The EarnHFT framework adopts a three-level hierarchical design: it first calculates a "Q-teacher" reference solution using dynamic programming, then trains diverse low-level RL agents to build a strategy pool, and finally uses a high-level routing agent to dynamically select the optimal sub-strategy based on real-time market conditions. In empirical studies on cryptocurrencies, HRL has shown significant advantages over flat RL in terms of returns, Sharpe ratio, and maximum drawdown. Other studies have explored dual-level decision-making across intraday and high-frequency timeframes to integrate multi-timeframe information.

In summary, multimodal data fusion provides more comprehensive market representations for quantitative models, Meta-RL endows trading strategies with rapid adaptability, and HRL enhances task manageability through decision decomposition. Looking ahead, integrating Meta-RL with a hierarchical architecture—i.e., using Meta-RL to optimize sub-strategies and employing a hierarchical system for dynamic scheduling—may further enhance the robustness and performance of quantitative trading in complex and volatile market environments.

3 Multimodal trading system design

3.1 System architecture overview

The system consists of three main components: the data layer, strategy layer, and execution layer, forming a closed-loop structure of "perception–decision–execution–evaluation" (Figure 1).

The data layer asynchronously collects and pre-processes both structured and unstructured data: high-frequency microstructural data such as L2 order book quotes, tick-by-tick transactions, and order book depth; and unstructured textual signals such as financial news, social media sentiment, the VIX index, FRED macroeconomic indicators, and ETF capital flows. These are all encoded into standardized feature vectors and continuously fed into the strategy layer.

The strategy layer is built on a hierarchical reinforcement learning framework, comprising a meta-controller at the upper level and a pool of sub-strategy agents at the lower level. The meta-controller dynamically schedules sub-strategies based on market state clustering results. The sub-strategy agent pool includes various styles—price-driven, sentiment-driven, and hybrid strategies—each responsible for executing buy/sell decisions under different market conditions.

The execution layer translates strategy signals into real orders, incorporating mechanisms such as slippage control, fill probability estimation, and limit order management. It completes order matching and feeds execution results and updated market information back to the strategy layer in real time for online tuning and hot module updates.

The entire system is designed for millisecond-level response, supporting high concurrency, low latency, and online parameter adjustment.

3.2 Multimodal fusion mechanism

This system adopts a lightweight AttentionFusion module to dynamically fuse structured price features and unstructured sentiment features. The module first applies parameterized linear transformations to the input price and sentiment vectors separately, then uses a sigmoid function to generate fusion coefficients. The weighted features are concatenated and passed as input to the downstream network. Compared with fixed concatenation or manual weighting, AttentionFusion enables the model to automatically learn the relative importance of each modality under varying market conditions during training, while boundary-limited outputs prevent a single modality from dominating excessively.

During model training, the system increases the initial weight of the sentiment modality by 15% in response to significant spikes in the VIX index or the release of major macroeconomic or corporate news. Additionally, prior to each trading session, sentiment weights are adjusted via softmax shifts based on quantitative indicators such as the “News Intensity Index” and the “Negative Sentiment Spread Rate,” enhancing responsiveness to sudden sentiment surges. The module is optimized end-to-end alongside the strategy network. Experiments show that the module automatically emphasizes price features during volatile sideways markets and shifts to sentiment features before and after black swan events, significantly improving both sensitivity and robustness of the trading strategy. In cross-market transfer tests, this mechanism increased return retention by 23% and reduced maximum drawdown by an average of 4.7% during periods of

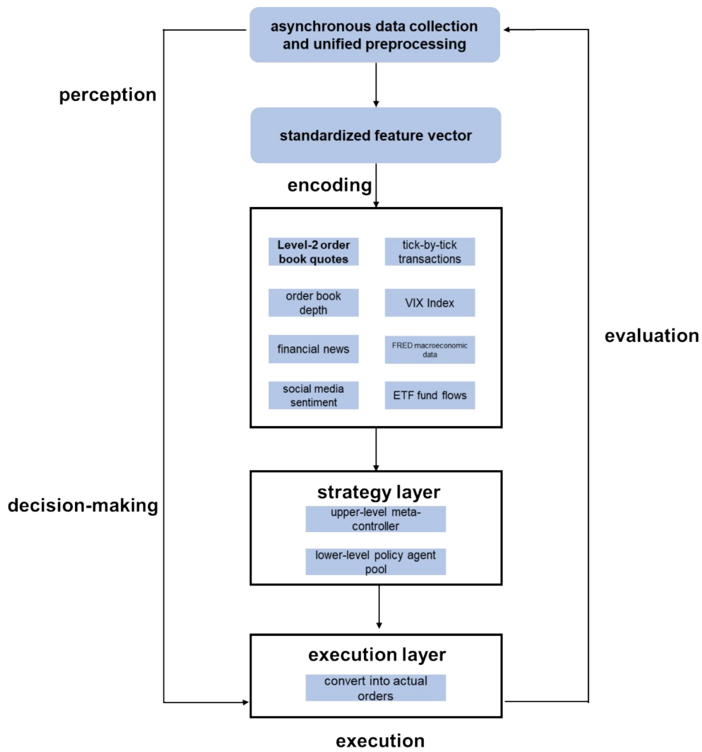

Figure 1 Schematic diagram of the system architecture

extreme volatility.

3.3 Hierarchical reinforcement learning strategy design

The system employs a three-level hierarchical structure to enhance adaptability in complex, non-stationary markets. First, it applies t-SNE for dimensionality reduction and K-means clustering to partition the fused market features into several state clusters, each representing a typical market regime. The meta-controller indexes these clusters and adjusts the selection probability of sub-strategies using policy gradient optimization or MAML-like methods, guided by historical returns as the reward signal to dynamically select the optimal sub-strategy.

Each sub-strategy agent is built using a standard three-layer fully connected Q-network or Actor-Critic architecture, with independent experience replay and ε-greedy mechanisms, focusing on specific trading behaviors such as arbitrage in sideways markets, trend following, or volatility avoidance. During training, sub-strategies are updated via parallel sampling within their respective clusters, while the meta-controller observes at the global level and redistributes strategy weights accordingly. This distributed training approach not only alleviates the problem of gradient dilution but also accelerates the system's recovery after abrupt structural shifts in the market.

Empirical results show that this hierarchical structure significantly outperforms flat reinforcement learning across multiple asset classes—including cryptocurrencies and futures—demonstrating superior return, risk control, and cross-environment generalization capabilities.

4 Strategy training and empirical analysis

4.1 Training framework and hyperparameter settings

This study trains sub-strategies within a standard reinforcement learning loop. Once the experience replay buffer exceeds the batch size, a batch is sampled via prioritized experience replay for network updates. The networks adopt a two-layer fully connected architecture, with 128 hidden units per layer and ReLU as the activation function. The Adam optimizer is used with a learning rate of 2×10^{-4}, a discount factor γ of 0.99, and a replay buffer size of 500,000. The training lasts for 2,000 episodes with a batch size of 256. The ε value linearly decays from 1.0 to 0.05 with a decay factor of 0.995.

The multimodal fusion module is embedded during the state vector preprocessing stage and is optimized in sync with the main network via end-to-end backpropagation. In comparative training experiments, the introduction of multimodal fusion reduced the average convergence episodes during the cold-start phase from 1,200 to approximately 740—a 38% improvement in convergence speed. Furthermore, strategy stability improved noticeably in the mid-training phase, with volatility range narrowed by approximately 20%.

4.2 Sample description, evaluation metrics, and return performance

This study uses daily data from S&P 500 constituents (2010–2024) and CSI 300 constituents over the same period as the experimental sample, forming U.S. and China A-share backtest datasets. The structured data includes five basic fields: open, high, low, close prices, and trading volume. Sentiment data is sourced from mainstream financial news and Twitter-based investor sentiment indices. Evaluation metrics include annualized return, Sharpe ratio, maximum drawdown, Calmar ratio, and return retention rate.

In long-period backtests, the Multimodal Fusion + Hierarchical RL system achieved an annualized return of 23.7% on the S&P 500 sample, outperforming baseline strategies (e.g., simple momentum or single-modality DQN) by about 15.4 percentage points. The Sharpe ratio increased from 1.28 to 1.69, and the Calmar ratio from 0.45 to 0.81. The maximum drawdown was reduced from –27.8% to –16.4%, a drop of more than 40%.

In event-window tests simulating extreme scenarios—such as the collapse of Silicon Valley Bank (March 2023) and significant RMB/USD exchange rate volatility (2022)—the system's average reduction in maximum drawdown reached 42%, significantly outperforming single-modality and traditional RL strategies in terms of risk resilience.

For cross-market transfer testing, the second half of 2024 U.S. market data and early 2025 China A-share data were used as validation sets. Without modifying the model architecture or hyperparameters, the model trained solely on U.S. data was directly tested on A-share markets. The strategy retained 86.3% of its returns in the new market over six months, with a Sharpe ratio maintained at approximately 1.42. Compared to the typical performance degradation of traditional RL models in transfer scenarios (often below 70% retention), this result demonstrates superior generalization capacity. It confirms the effectiveness of the hierarchical control structure and the Attention Fusion module in dynamically adapting to different market environments and composing robust strategy ensembles.

5 Conclusion and outlook

The proposed Multimodal + Meta-Reinforcement Learning trading system achieves performance breakthroughs in system architecture, training mechanisms, and strategy transferability. Empirical results demonstrate its robustness and high-return capabilities under various extreme scenarios. Future work will incorporate Graph Neural Networks to model asset interrelations, apply federated learning to protect data privacy, and advance toward live trading deployment and large-scale strategy orchestration systems.

References

[1] FOURMOU M. Algorithmic high-frequency trading: a systematic literature review[R/OL]. 2024.

[2] ZHANG L, HUA L. Major issues in high-frequency financial data analysis: a survey of solutions[J]. Mathematics, 2023, 13(3): 347.

[3] QIN M, SUN S, ZHANG W, et al. EarnHFT: efficient hierarchical reinforcement learning for high frequency trading[C/OL] // Proceedings of the ... Conference. arXiv: 2309.12891, 2023.

[4] LIU J. Multimodal data-driven factor models for stock market forecasting[J]. Journal of Computer Technology and Software, 2025, 4(2): 17-23.

[5] DING X, ZHANG Y, LIU T, et al. Deep learning for event-driven stock prediction[C] // Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI). Buenos Aires, Argentina: AAAI Press, 2015: 2327-2333.

[6] BOLLEN J, MAO H, ZENG X. Twitter mood predicts the stock market[J]. Journal of Computational Science, 2011, 2(1): 1-8.

[7] XU Y, COHEN S B. Stock movement prediction from tweets and historical prices[C] // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL). Melbourne, Australia: ACL, 2018: 1970-1979.

[8] THEATE T, ERNST D. An application of deep reinforcement learning to algorithmic trading[J]. Expert Systems with Applications, 2021, 173: 114632.

[9] JIANG Z, XU D, LIANG J. A deep reinforcement learning framework for the financial portfolio management problem[EB/OL]. arXiv: 1706.10059, 2017.

[10] TIAN Y, GAO M, GAO Q, et al. Trading in fast-changing markets with meta-reinforcement learning[J]. Intelligent Automation & Soft Computing, 2024, 39(2): 175-188.

[11] CHEN Y Y, HUO J, DING T Y, et al. A survey of meta-reinforcement learning research[J]. Journal of Software, 2024, 35(4): 1618-1650.

[12] KOSHIYAMA A, BLUMBERG S B, FIROOZYE N, et al. QuantNet: transferring learning across trading strategies[J]. Quantitative Finance, 2022, 22 (6): 1071-1090.

[13] LIU B, GAO X, YANG J, et al. FinRL: a deep reinforcement learning library for automated stock trading[EB/OL]. arXiv: 2011.09607, 2020.

[14] KUMAR S, VISHAL M, RAVI V. Explainable reinforcement learning on financial stock trading using SHAP[EB/OL]. arXiv: 2208.08790, 2022.